2. Material and Method

2.1 Machine learning model building

This work was conducted using the dataset of HCC. The dataset includes N=217 patients diagnosed with Hepatocellular Carcinoma (HCC). A detailed description of heterogenous dataset comparing 16 quantitative features and 9 qualitative features (n=16+9=25) is shown in the next table.Split the data into training and testing, the training data was 173 samples, and the testing data was 44 samples. Here is the flow diagram of our steps:

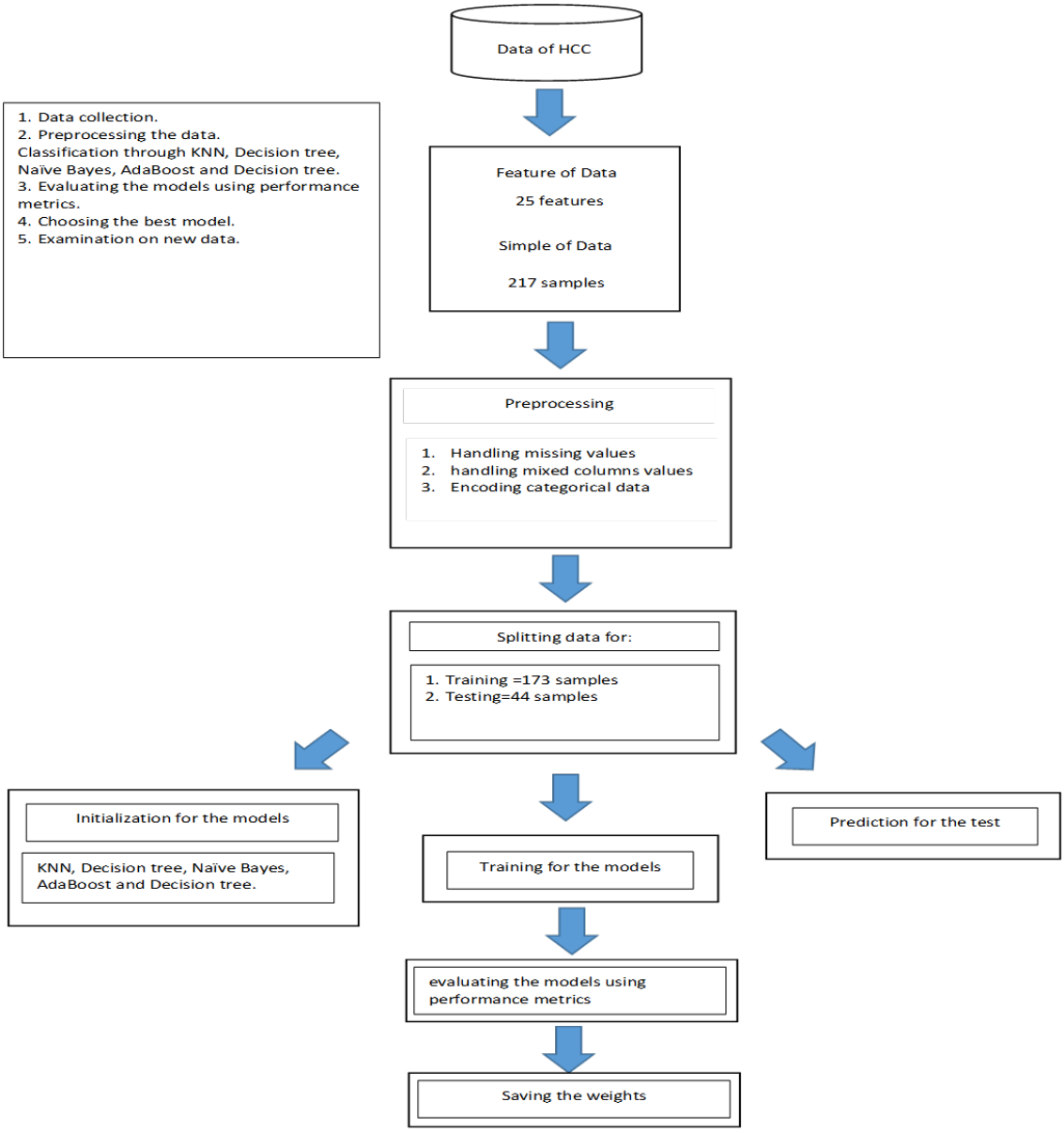

Figure 1. The flow diagram shows the main steps of machine learning, data processing and model training.

Quantitative Features	Mean	StdDev
RQmiR129	4.819889	11.051431
RQmiR1262	4.711627	11.821789
RQmiR106b	26.664873	114.485184
RQmRNARAB11	2.018319e+04	2.188484e+05
RQSTAT	0.686805	0.500463
RQmRNAATG	208.765081	789.62555
RQLncWRAP	1633.239835	19154.465853
RQLncRNARP11513115	40.992973	84.808809
Age	56.949074	6.548012
AST	27.146296	40.807424
ALT	56.580556	36.510439
T.bilirubin	6.849833	14.129344
D.bilirubin	1.863472	2.384609
Albumin	2.356481	2.099473
INR	2.313426	0.969085
AFP	120.475047	407.303333

Table 1. shows statistical summarization of each quantitative features.

2.2 Data preprocessing and normalization

The dataset has 3 classes. Class 1 if a patient has Benign Liver condition (68 records as class 1), Class 0 if a patient has Malignant HCC (102 records labeled as class 0), Class 2 if a patient is healthy (47 records labeled as class 2). The last 2 rows had been removed as their values are NAN, which is not necessary. The last 3 columns have lack of information, they don't have any information about the case if it's Benign or Healthy. So, it's better to drop these columns. The data has 20 missing values, so it has been filled using the mode of each column. The HCV.ABS column has mixed datatypes of strings and floats, so the string values had been converted to numerical. Applied encoding to the categorical features, then encoding the label. Applied Standardization (Z-score) on the data. The values were normalized based on the mean and standard deviation (SD) as follows **Z (Z-score)** = $\frac{X-\mu}{\sigma}$, where X is the feature value, μ is mean value and σ is SD, this allows us to standardize data so that observations can be compare from different data sets that may have different means and standard deviations. A z-score of 0 indicates that the observation is equal to the mean, while a positive z-score indicates that the

observation is above the mean, and a negative z-score indicates that the observation is below the mean.
then, Splitting the features and the label from each other.

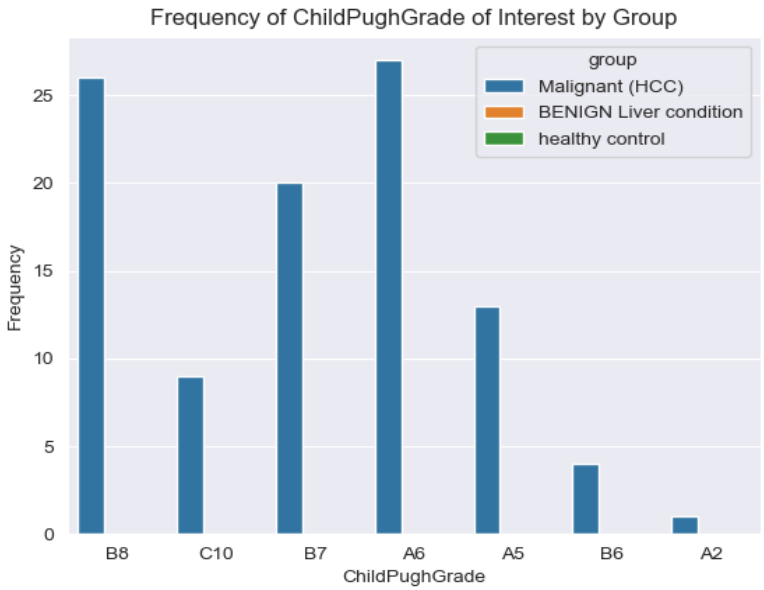

Figure2. Frequency of childPughGrade of interest by group, it is about a count plot that calculates the frequency of each value in the column.

The statistical equation for a count plot can be summarized as follows:

- Divide the data set into a set of categories.
- Count the number of observations in each category to obtain the frequency of observations in each category.
- Display the frequency of observations in each category using bars of equal width, with the height of each bar corresponding to the frequency of observations in the corresponding category.

As shown in the graph, each value has been existed for class Malignant only in range ~(3 – 28), in benign and healthy, it doesn't exist at all, so it has been dropped, the (blue color) represents Malignant, (orange color) represents Benign and (the green) color represents healthy.

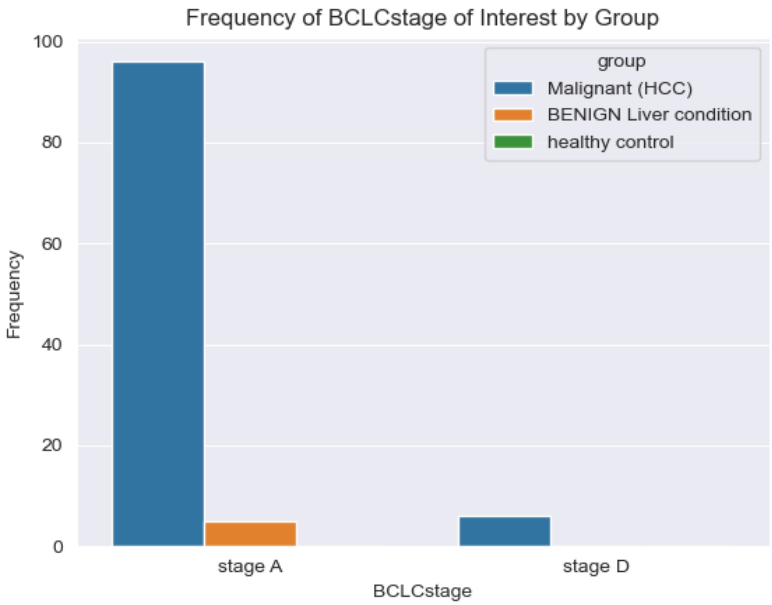

Figure3. Frequency of BCLCstage of interest by group, it is about a count plot, its statistical equation as shown in Figure2. every stage has been existed for class Malignant in range ~(8 – 90), in benign, stage A only was about 5 has been existed for class Benign, in healthy, the two stages don't exist at all, so it has been dropped, the (blue color) represents Malignant, (orange color) represents Benign and the (green color) represents healthy.

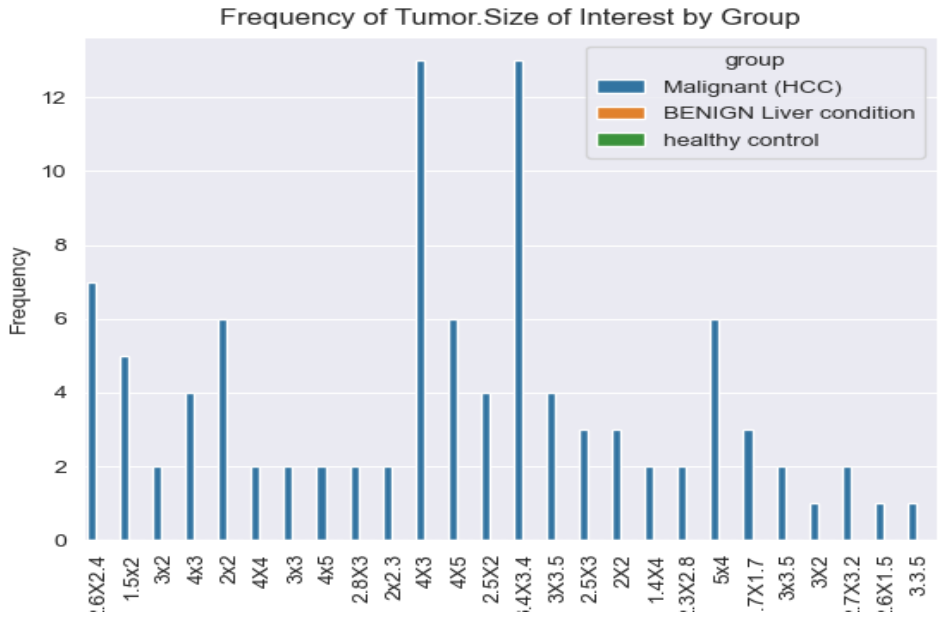

Figure4. Frequency of Tumor.Size of interest by group, it is about a count plot, its statistical equation as shown in Figure2. each value has been existed for class Malignant only in range ~(1 – 15), in benign and healthy, it doesn't exist at all, so it has been dropped, the (blue color) represents Malignant, (orange color) represents Benign and the (green color) represents healthy.

2.3 Correlation analysis and Feature selection

Feature selection is the process of selecting a subset of the most relevant features (or variables) from a larger set of features in a dataset. The goal of feature selection is to reduce the dimensionality of the data and improve the performance of machine learning models by removing irrelevant or redundant features that may lead to overfitting or increase the computational complexity of the model.

There are several approaches to feature selection, including filter methods, wrapper methods, and embedded methods.

Some popular feature selection algorithms include Principal Component Analysis (PCA), Recursive Feature Elimination (RFE), SelectKBest, and L1-based methods such as Lasso and Elastic Net.

Feature selection is an important step in machine learning pipelines, especially when dealing with high-dimensional datasets. It can help to improve the accuracy, interpretability, and efficiency of machine learning models, and reduce the risk of overfitting and data leakage.

CFS method had been used, CFS (Correlation-based Feature Selection) is a filter method for feature selection that evaluates the relevance and redundancy of each feature based on their

correlations with the target variable and with each other. CFS is based on the idea that the best subset of features is the one that maximizes the relevance (correlation) with the target variable while minimizing the redundancy (correlation) among the features.

The CFS algorithm consists of the following steps:

- Compute the correlation coefficient between each feature and the target variable.
- Compute the correlation coefficient between each pair of features.
- Normalize the correlation coefficients to have values between 0 and 1.
- Compute the merit score of each feature as the difference between its average correlation with the target variable and its average correlation with all other features.
- Select the top k features with the highest merit scores.
- The idea behind the merit score is to select features that are highly correlated with the target variable but poorly correlated with other features, as they provide unique information and reduce redundancy in the feature space.
- CFS is a fast and simple method for feature selection that can handle large datasets with high-dimensional feature spaces. It has been shown to outperform other filter methods such as correlation-based and mutual information-based methods in terms of accuracy and efficiency.

2.3.1 Feature-feature correlation for the whole available data set using Pearson correlation.

Feature-feature correlation is a statistical measure that evaluates the linear relationship between pairs of features in a dataset. Pearson correlation coefficient is a commonly used measure of feature-feature correlation that ranges from -1 to 1, where -1 indicates a perfect negative correlation, 0 indicates no correlation, and 1 indicates a perfect positive correlation.

To compute the feature-feature correlation for the whole available dataset using Pearson correlation, compute the Pearson correlation coefficient between each pair of features in the dataset.

2.3.2 Feature-feature correlation for the whole available data set using Fisher correlation.

Feature-feature correlation using Fisher correlation is a statistical measure that evaluates the linear relationship between pairs of features in a dataset using the Fisher transformation. Fisher transformation is a method to transform the Pearson correlation coefficient into a more normally distributed variable that can be used for hypothesis testing.

To compute the feature-feature correlation for the whole available dataset using Fisher correlation, compute the Fisher transformation of the Pearson correlation coefficient between each pair of features in the dataset.

2.3.3 Feature-class correlation (Feature Importance Ranking)

Feature-class correlation, also known as feature importance ranking, is a technique for evaluating the relevance of each feature in a dataset for predicting the target variable or class. The goal of feature importance ranking is to identify the most informative features that contribute the most to the predictive power of machine learning models.

2.4 multi-class classification

2.4.1 Random Forest:

Random Forest is a popular and powerful machine learning algorithm used for classification, regression, and other tasks. It is a type of ensemble learning method that combines multiple decision trees to make predictions.

Random Forest is a popular machine learning algorithm for classification problems, especially when dealing with a relatively small dataset like this dataset. Here are some reasons why Random Forest might have been the best choice for HCC dataset:

Handling high-dimensional data: Random Forest can handle a large number of input features or variables, which is common in medical datasets. The algorithm builds multiple decision trees using a random subset of features, which helps to avoid overfitting and improve the performance of the model.

Handling imbalanced data: Medical datasets often have class imbalances, where one class (e.g., healthy patients) has many more samples than another class (e.g., patients with HCC). Random Forest can handle imbalanced data by using a technique called "bagging" or bootstrap aggregating, which resamples the data to balance the classes and reduces the variance of the model.

Robustness to noise: Medical datasets can have noise or outliers due to various reasons. Random Forest is a robust algorithm that can handle noisy data by averaging the predictions of multiple decision trees.

Ease of use and tuning: Random Forest is easy to implement and tune, and it can provide good performance with minimal hyperparameter tuning. The algorithm has only a few hyperparameters to tune, such as the number of trees, the maximum depth of the trees, and the number of features to consider at each split.

Good performance on many datasets: Random Forest has been shown to perform well on many different types of datasets, including medical datasets. This is because it is a versatile algorithm that can handle different types of data and can be applied to both binary and multi-class classification problems.

2.4.2 KNN:

KNN is a simple and effective algorithm for classification that works by finding the K nearest neighbors of a new data point based on a distance metric (e.g., Euclidean distance). The algorithm then assigns the class label of the majority of the K neighbors to the new data point.

The algorithm would first be trained on a subset of the data, using the features as input and the class labels as output. Then, for a new patient, the algorithm would compute the distances to the K nearest neighbors in the training set and assign the class label based on the majority vote of those neighbors.

KNN can be a good choice for this dataset as the number of features is relatively small. However, KNN can be sensitive to the choice of the distance metric, the number of neighbors, and the scaling of the input features. It can also be computationally expensive for large datasets or high-dimensional data.

2.4.3 Naïve Bayes:

Naive Bayes is a popular machine learning algorithm for classification problems that is based on the Bayes theorem and the assumption of conditional independence between the input features given the class label.

The Naive Bayes model predicts the class label of a new data point based on the probability of the input features given each class label, and the prior probability of each class label. The model assumes that the input features are conditionally independent given the class label, which simplifies the calculation of the likelihood.

The Naive Bayes model can be trained using maximum likelihood estimation or maximum a posteriori estimation, which involves estimating the parameters of the likelihood and the prior probability from the training data. This can be done using simple counting or smoothing techniques such as Laplace smoothing or Lidstone smoothing. The trained model can then be used to predict the class labels of new data points based on their input features.

2.4.4 AdaBoost:

AdaBoost, short for Adaptive Boosting, is a machine learning algorithm that is used for classification and regression problems. It is an ensemble learning method that combines multiple weak learners (i.e., simple models that perform only slightly better than random guessing) to form a strong learner that can make accurate predictions.

AdaBoost has several advantages as a machine learning algorithm:

It can achieve high accuracy with relatively simple weak learners.It is less prone to overfitting than some other algorithms, especially when the weak learners are not too complex.It can handle datasets with noisy or irrelevant features by giving them lower weights.It can be used for binary

and multi-class classification problems, as well as for regression problems.However, AdaBoost also has some limitations:It can be sensitive to outliers or noisy data points that are misclassified by the weak learners.It can be computationally expensive, especially when the number of weak learners is large, or the dataset is large. It can be difficult to interpret the final model and explain its results to non-experts.

2.4.5 Decision Tree:

Decision trees are a popular machine learning algorithm for classification and regression problems that can be used for your HCC liver dataset. Here's a brief description of decision trees and their use on your data that you can share with your client:

A decision tree is a model that represents a sequence of decisions that lead to a final prediction or decision. The tree consists of nodes that represent the input features, branches that represent the possible values of each feature, and leaves that represent the class labels or regression values. The decision tree algorithm works by recursively splitting the data into subsets based on the input features that best separate the classes or minimize the error in regression problems.

In the context of your HCC liver dataset, decision trees could be used to predict whether a patient has HCC or not based on their clinical and laboratory data. The algorithm would first be trained on a subset of the data, using the features as input and the class labels as output. Then, for a new patient, the algorithm would follow a sequence of decisions based on the input features to arrive at a prediction of the class label.

Decision trees can be a good choice for your dataset if the data has clear boundaries between the classes and the relationships between the input features and the class label are non-linear or complex. Decision trees can also be easy to interpret and explain to non-experts, and they can handle missing values and noisy data.

To use decision trees on your data, you would need to preprocess the data by splitting it into training and test sets and handling missing values and categorical features appropriately. You could then train the decision tree algorithm on the training set using an appropriate splitting criterion (e.g., Gini impurity or information gain), and a stopping criterion that prevents overfitting (e.g., maximum depth or minimum number of samples per leaf). The trained model could then be evaluated on the test set using appropriate evaluation metrics such as accuracy, precision, recall, and F1-score.

It's important to note that decision trees can suffer from overfitting if the tree is too complex or if the data is noisy or contains irrelevant features. To address this, you could use techniques such as pruning, ensemble methods (e.g., Random Forest), or regularization to improve the generalization performance of the model.

In summary, decision trees are a powerful and interpretable machine learning algorithm that can be used for classification and regression problems. To use decision trees on your HCC liver dataset, you would need to preprocess the data, train the decision tree algorithm on the training set, and evaluate its performance on the test set using appropriate evaluation metrics. Decision trees can be a good choice if the data has non-linear or complex relationships between the input features and the class label, and if interpretability is important.

2.5 Cross Validation

K-fold cross validation had been used. K-Fold Cross Validation is a technique for validating the performance of a machine learning model by partitioning the dataset into K equal-sized subsets, called folds, and training and testing the model K times, each time using a different fold for testing and the remaining K-1 folds for training.

The K-Fold Cross Validation algorithm works as follows:

1. Split the dataset into K equal-sized folds.
2. For each fold i in 1, 2, ..., K, do the following:
    a. Use the i-th fold as the testing set and the remaining K-1 folds as the training set.
    b. Train the machine learning model on the training set and evaluate its performance on the testing set using a specific performance metric (e.g., accuracy, F1-score, etc.).
    c. Record the performance metric for fold i.
3. Compute the average performance metric across all folds.

The K-Fold Cross Validation algorithm helps to reduce the variance of the performance estimate by using multiple testing sets and averaging the results. It also helps to avoid overfitting by using different subsets of the data for training and testing.

In practice, the value of K is typically set to 5 or 10, but can be adjusted depending on the size and complexity of the dataset. A higher value of K can lead to a more accurate estimate of the performance, but also requires more computational resources and time.

2.6 PCA analysis

PCA (Principal Component Analysis) is a dimensionality reduction technique that transforms a high-dimensional dataset into a lower-dimensional representation while preserving the most important information in the data. PCA is a method commonly used in data analysis and machine learning to reduce the number of features in a dataset, while retaining as much of the original information as possible.

PCA works by finding the principal components of the data, which are the directions in the original feature space that capture the most variance in the data. The first principal component is the direction that explains the largest amount of variance in the data, and each subsequent

principal component explains the largest amount of variance that is orthogonal (uncorrelated) to the previous components. The number of principal components is equal to the number of dimensions in the original dataset.

2.7 Evaluation of Machine Learning

The dataset has 217 patients. The test dataset was used to evaluate model performance. The model was compared on key performance criteria that included ROC curve, a 2 × 2 confusion matrix was performed and a classification report.

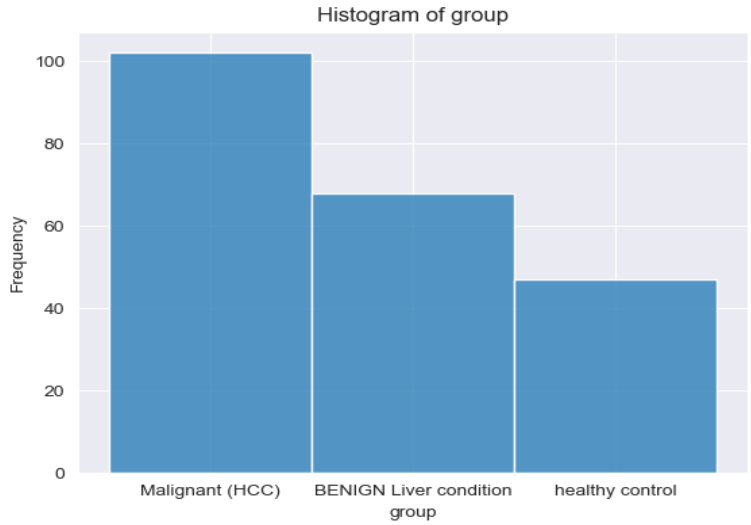

Figure5 displays the frequency of each class, it is a histogram that displays the frequency of group column, the statistical equation for a histogram can be summarized as follows:

- Dividing the range of the data into a set of intervals or bins of equal width.
- Counting the number of observations in each interval to obtain the frequency of observations in each interval.
- Calculating the relative frequency of observations in each interval by dividing the frequency by the total number of observations in the data set.
- Displaying the frequency or relative frequency of observations in each interval using bars of equal width, with the height of each bar corresponding to the frequency or relative frequency of observations in the corresponding interval.

As shown in the graph, the malignant class frequency was about 102, benign class frequency was 68 and healthy class frequency was 47.

3. Results

3.1 Machine Learning Screening

K-fold cross validation had been used with number of k=5, at each iteration the data is split into 174 training samples and 43 testing samples.

One vs. all approach had been used as the problem is multi class classification. One-vs-all (also known as one-vs.-rest) is a popular technique to extend binary classification algorithms to multi-class problems. The idea behind one-vs-all is to train a separate binary classifier for each class, which distinguishes that class from the other classes. The classifiers are trained by treating all the instances of a particular class as positive instances and the instances of all other classes as negative instances. During prediction, each classifier is applied to the input instance, and the class associated with the classifier that gives the highest prediction score is chosen as the predicted class.

Each model had been trained and evaluated it before applying feature selection and PCA and train the best 2 models after applying the feature selection and PCA.

3.2 Feature Selection

Feature-feature correlation had been applied for the whole available data set using Pearson correlation and then, applied Feature-feature correlation for the whole available data set using Fisher correlation. The important features had been chosen based on the CFS method. After that PCA was applied on the selected features. First the data had been explored and relationships between its features using visualization plots.

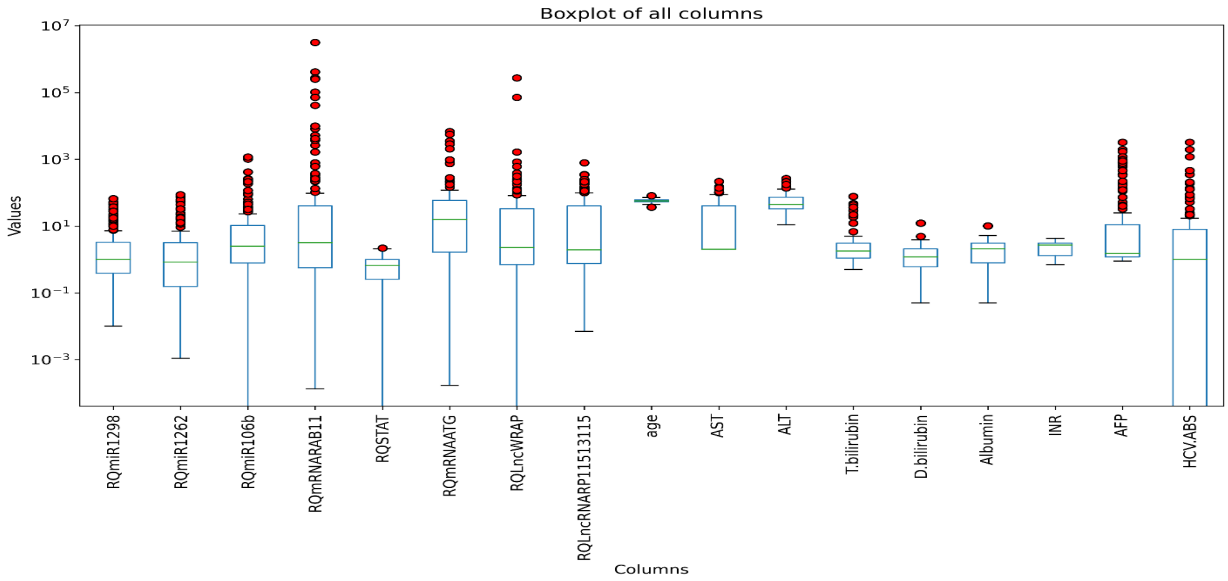

Figure6.displays the box plot of each column, it is a graphical representation of a data set that summarizes its distribution. The box in the plot represents the interquartile range (IQR), which is the range of values that fall between the 25th and 75th percentiles of the data set. The median is shown as a line inside the box. The "whiskers" extending from the box indicate the range of the data set outside of the IQR, up to a certain distance from the box, typically 1.5 times the IQR, the (green line) represents the median of the data and the (red points) are outliers, the RQmRNARAB11 has the most outliers as shown in the plot, outliers are removed using Z-score.

The statistical equation for a box plot can be broken down into several components:

- Quartiles (Q1, Q2, Q3): The box in a box plot represents the middle 50% of the data, with the bottom and top edges of the box corresponding to the first and third quartiles, respectively. The second quartile (Q2) is the median of the data set.
- Interquartile range (IQR): The IQR is the range of the middle 50% of the data and is calculated as the difference between the third and first quartiles: IQR = Q3 - Q1.
- Whiskers: The whiskers in a box plot extend from the top and bottom edges of the box to the largest and smallest observations within 1.5 times the IQR of the box. Any observations outside of this range are considered outliers and are usually displayed as individual points.

The statistical equation for a box plot can be summarized as follows:

- Lower whisker: Q1 - 1.5 x IQR
- Lower edge of the box: Q1
- Median (middle line of the box): Q2
- Upper edge of the box: Q3
- Upper whisker: Q3 + 1.5 x IQR

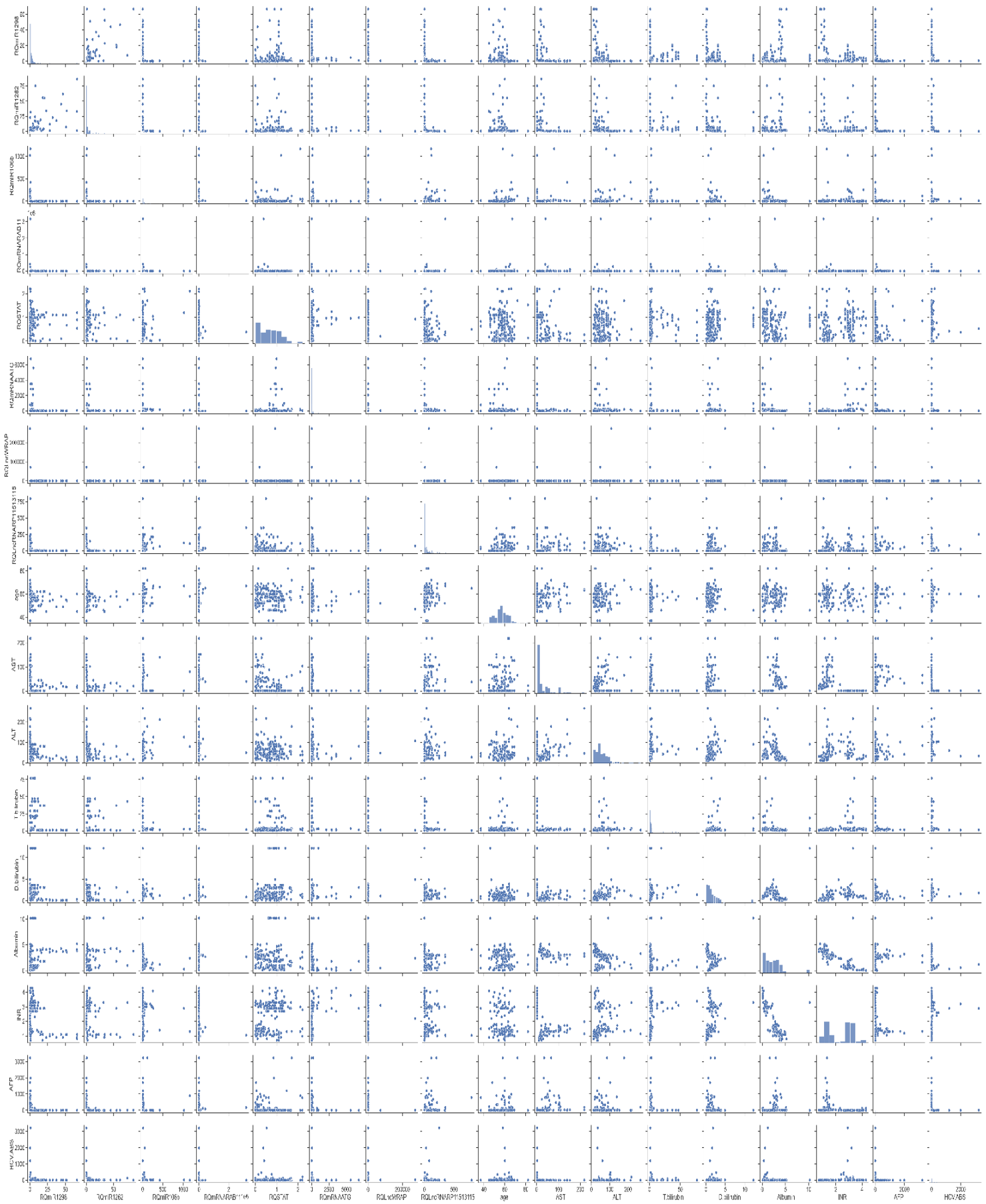

Figure7.displays a visualization of the pairwise relationships between different variables in a dataset to quickly identify any patterns or relationships between variables in dataset.

The statistical equation for a pairwise plot can be broken down into several components:

- Scatterplots: A scatterplot is a plot of two variables against each other, with one variable plotted on the x-axis and the other plotted on the y-axis. Each point on the scatterplot represents a single observation in the data set.
- Correlation coefficients: The correlation coefficient is a measure of the strength and direction of the linear relationship between two variables. It ranges from -1 to 1, with values closer to -1 indicating a strong negative correlation, values closer to 1 indicating a strong positive correlation, and values close to 0 indicating little or no correlation.
- Diagonal plots: The diagonal plots in a pairwise plot display the distribution of each variable in the data set. These plots in these data are histograms.

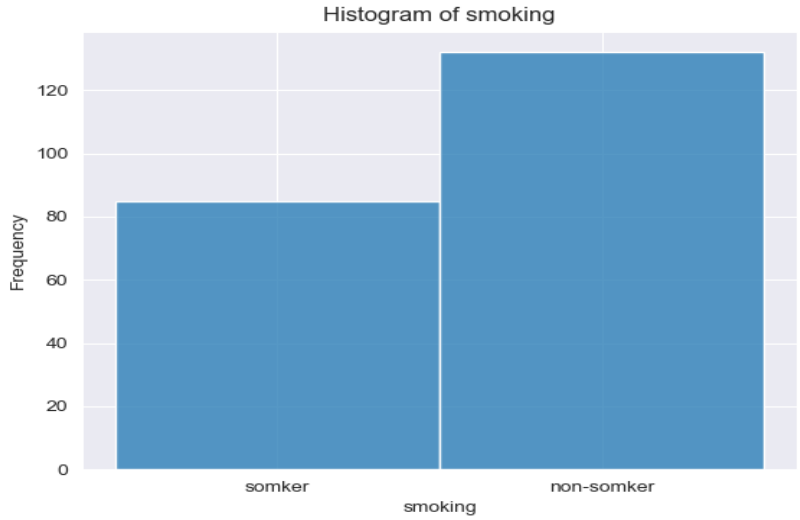

Figure8. Histogram that displays the frequency of smoking column, The statistical equation for a histogram as shown in Figure5. As shown in the graph, smoker frequency is about 83 and non-somker frequency is about 130.

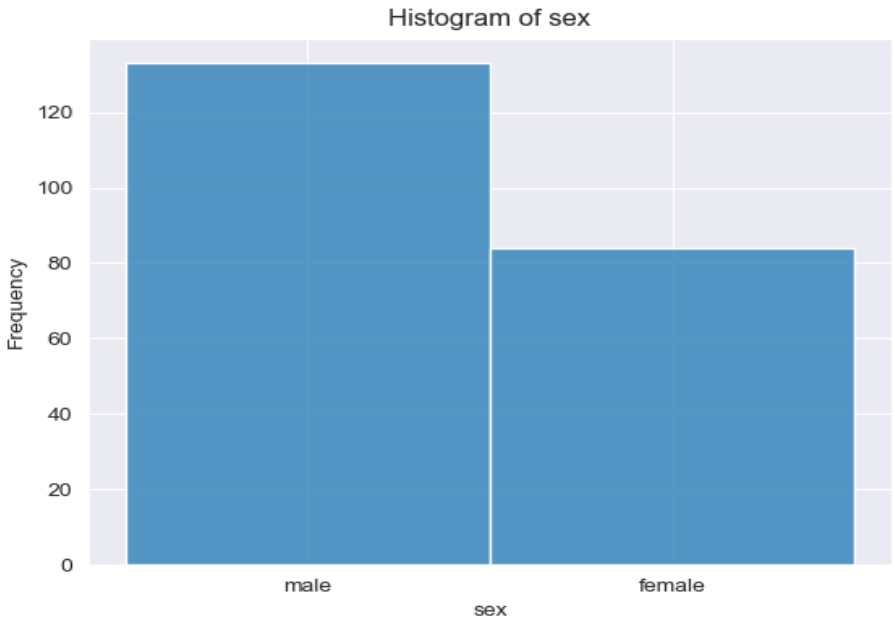

Figure9. Histogram displays the frequency of sex column, the statistical equation of histogram as shown in Figure5. As shown from the graph, male frequency is about 130, and female frequency is about 85.

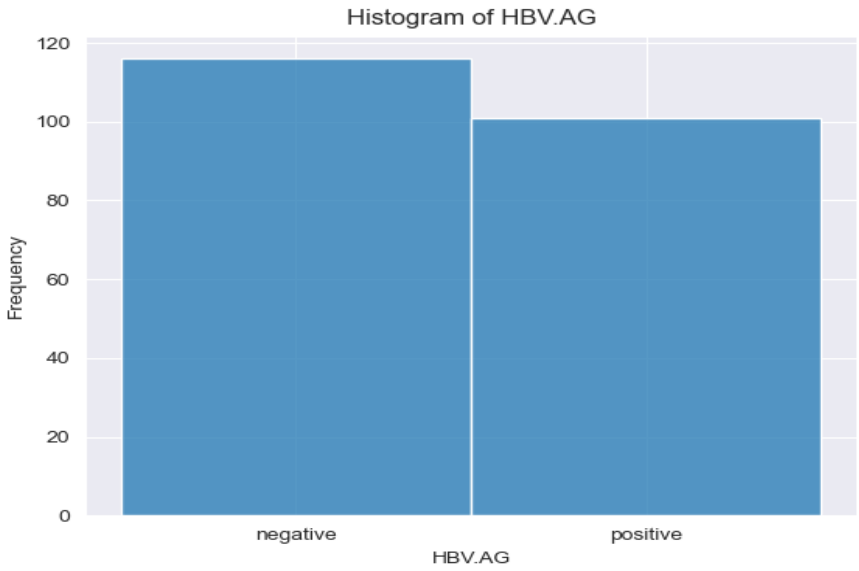

Figure10.displays the frequency of HBV.AG. The statistical equation of histogram as shown in Figure5. As shown from the graph, negative frequency is about 115, and positive frequency is about 102.

To get feature to feature correlation, first get the correlation matrix. A correlation matrix is a table that shows the correlation coefficients between pairs of variables in a data set. Correlation coefficients are a measure of the strength and direction of the linear relationship between two variables. A correlation matrix can help to identify patterns and relationships between variables and can be useful for exploratory data analysis and modeling. To create a correlation matrix, follow these steps:

- Collect the data: Gather the data for the variables you want to analyze.
- Calculate the correlation coefficients: used a statistical software package in Python to calculate the correlation coefficients between pairs of variables. The correlation coefficient can range from -1 to 1, with -1 indicating a perfect negative correlation, 1 indicating a perfect positive correlation, and 0 indicating no correlation.
- Create a table: Organize the correlation coefficients into a table, where each row and column represent a variable in the data set. The diagonal of the table should be filled with 1's since each variable is perfectly correlated with itself.
- Visualize the matrix: To make it easier to interpret the matrix, you can create a heatmap of the correlation coefficients. In the heatmap, colors can be used to represent the strength of the correlation, with darker colors indicating stronger correlations.
- It is important to note that correlation does not imply causation. A high correlation between two variables does not necessarily mean that one variable causes the other. Further analysis is needed to establish causality.

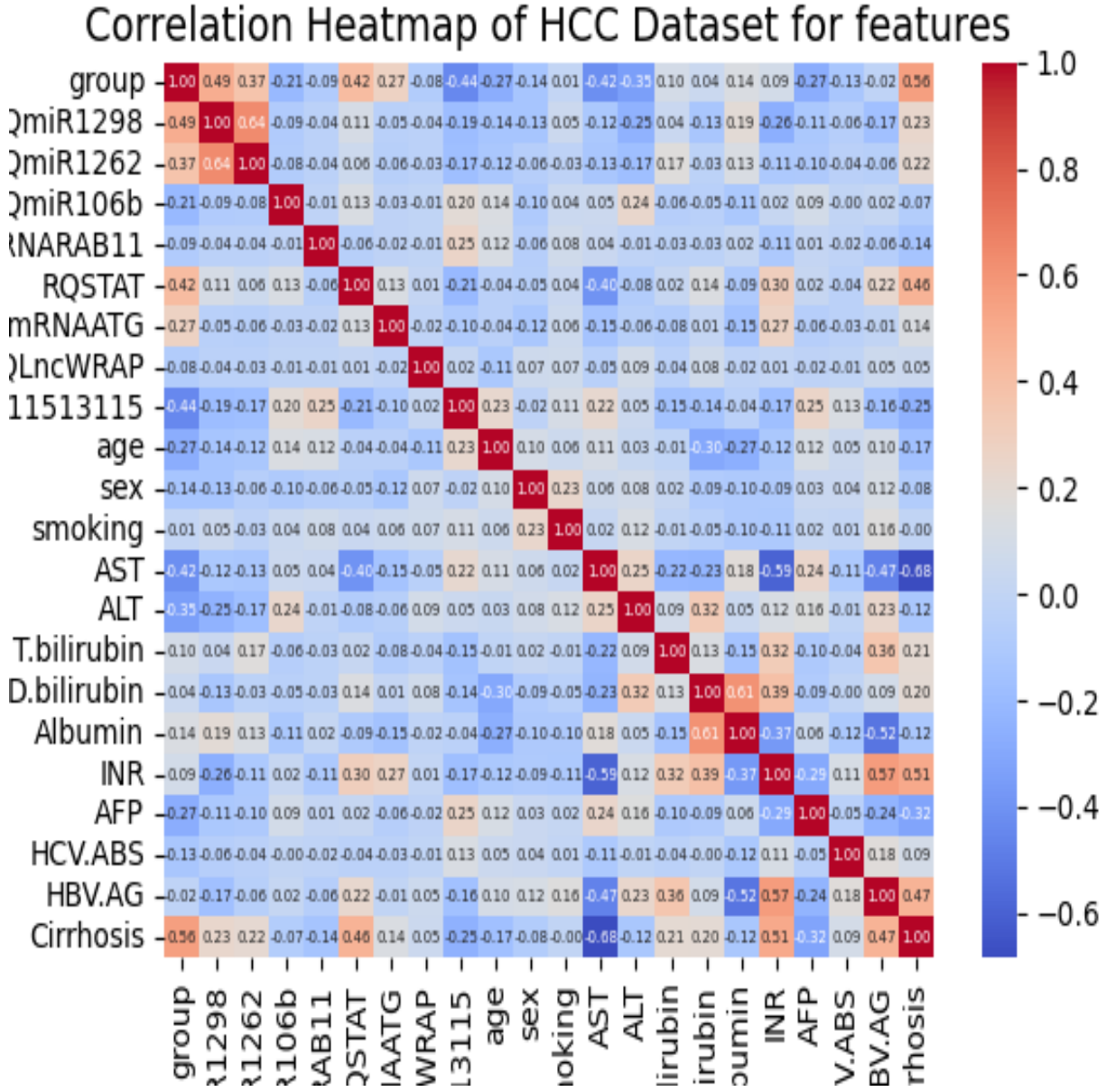

Figure11. displays the correlation Heatmap of HCC dataset features.

statistical equation for a heatmap can be broken down into several components:

- Data matrix: The data set is represented as a matrix, where each element in the matrix corresponds to a single observation in the data set.
- Color scale: A color scale is chosen to represent the values in the data set. Typically, the color scale ranges from a low value (blue) to a high value (red), with intermediate values represented by intermediate colors (e.g., pink, grey, orange..).

- Color mapping: The values in the data set are mapped to colors in the color scale. Higher values in the data set are typically represented by warmer colors (e.g., red, orange), while lower values are typically represented by cooler colors (e.g., blue, green).
- Heatmap display: The values in the data set are displayed as a grid of colored squares, where each square represents a single observation in the data set. The color of each square corresponds to the value of the corresponding observation, as determined by the color mapping.

Table.2 Feature to feature correlation table of Pearson correlation.

Feature to feature	Correlation degree	Degree significations
Cirrhosis	0.56	very strong
RQmiR1298	0.49	Very strong
RQSTAT	0.42	Very strong
RQmiR126	0.37	Very strong
RQmRNAATG	0.27	Very strong
Albumin	0.14	significant
T.bilirubin	0.10	intermediate
INR	0.09	Intermediate
D.bilirubin	0.04	Intermediate
Smoking	0.01	Intermediate
HBV.AG	-0.02	Intermediate
RQLncWRAP	-0.08	Intermediate
RQmRNARAB11	-0.09	Intermediate
HCV.ABS	-0.13	Significant
Sex	-0.14	Significant
RQmiR106b	-0.21	very strong
AFP	-0.27	very strong
Age	-0.27	very strong
ALT	-0.35	very strong
AST	-0.42	very strong
RQLncRNARP11513115	-0.44	very strong

The degree of significance refers to the probability that the correlation coefficient between two variables is not due to random chance. It is also known as the p-value of the correlation test.

The p-value is a measure of the strength of evidence against the null hypothesis, which is the hypothesis that there is no correlation between the two variables. A small p-value (typically less than 0.05) indicates that there is strong evidence against the null hypothesis, and that the observed correlation is unlikely to be due to chance. The degree of significance is usually reported alongside the correlation coefficient in statistical analyses, such as regression models or correlation matrices. It is important to consider both the magnitude of the correlation coefficient and the degree of significance when interpreting the results of a correlation analysis, as a high

correlation coefficient may not be meaningful if the degree of significance is low. If the p-value less than 0.05, the degree of significance is strong, if the p-value greater than 0.05, the degree of significance is intermediate, if the p-value less than 0.01, the degree of significance is very strong.

Tabel.3 Feature to feature correlation table of Fisher correlation.

Feature to feature	Correlation degree	Degree significations
Cirrhosis	0.63	intermediate
RQmiR1298	0.54	intermediate
RQSTAT	0.44	intermediate
RQmiR126	0.39	intermediate
RQmRNAATG	0.28	intermediate
Albumin	0.14	intermediate
T.bilirubin	0.10	intermediate
INR	0.09	Intermediate
D.bilirubin	0.04	Intermediate
Smoking	0.01	Intermediate
HBV.AG	-0.02	Intermediate
RQLncWRAP	-0.08	Intermediate
RQmRNARAB11	-0.09	Intermediate
HCV.ABS	-0.13	Significant
Sex	-0.14	Significant
RQmiR106b	-0.21	very strong
AFP	-0.28	very strong
Age	-0.28	very strong
ALT	-0.36	very strong
AST	-0.44	very strong
RQLncRNARP11513115	-0.48	very strong

Table5 of top features selected by Pearson and by Fisher Score (feature importance ranking) using CFS method.

Selected Feature	Pearson score	fisher score
RQmiR1298	0.49230037	0.22082227
RQmiR1262	0.37497906	0.12069712
RQSTAT	0.41550502	0.15091439
RQmRNAATG	0.27227125	
RQLncRNARP11513115	0.44297653	0.17389982
age	0.27494592	
AST	0.41539265	0.15082467
ALT	0.34693357	0.10216421
AFP	0.27182065	
Cirrhosis	0.5580884	0.29654732

Multi-Class Classification Results and evaluation

Here is the performance of each model with each class, trained them with all features, evaluated them using these metrics:

- Accuracy.
- F-score.
- Precision.
- Recall.

Table.6 Performance of each model with all features.

	KNN with k=5	Random Forest	Decision Tree with depth=10, min_samples_leaf=5, min_samples_split=0.5	Adaboost with n_estimators=10, learning rate=0.1	Naïve Bayes
Accuracy	0.82	0.96	0.86	0.91	0.95
F-score	0.81	0.95	0.84	0.89	0.95
Recall	0.83	0.95	0.83	0.9	0.95
Precision	0.82	0.96	0.86	0.88	0.95

Table7.Percentage accuracy of each class prediction

	Malignant	Benign	Healthy
KNN	0.96	0.75	0.71
Naïve Bayes	0.97	0.94	0.93
Random Forest	1.0	0.92	0.95
Decision Tree	0.92	0.83	0.74
Adaboost	0.95	0.84	0.95

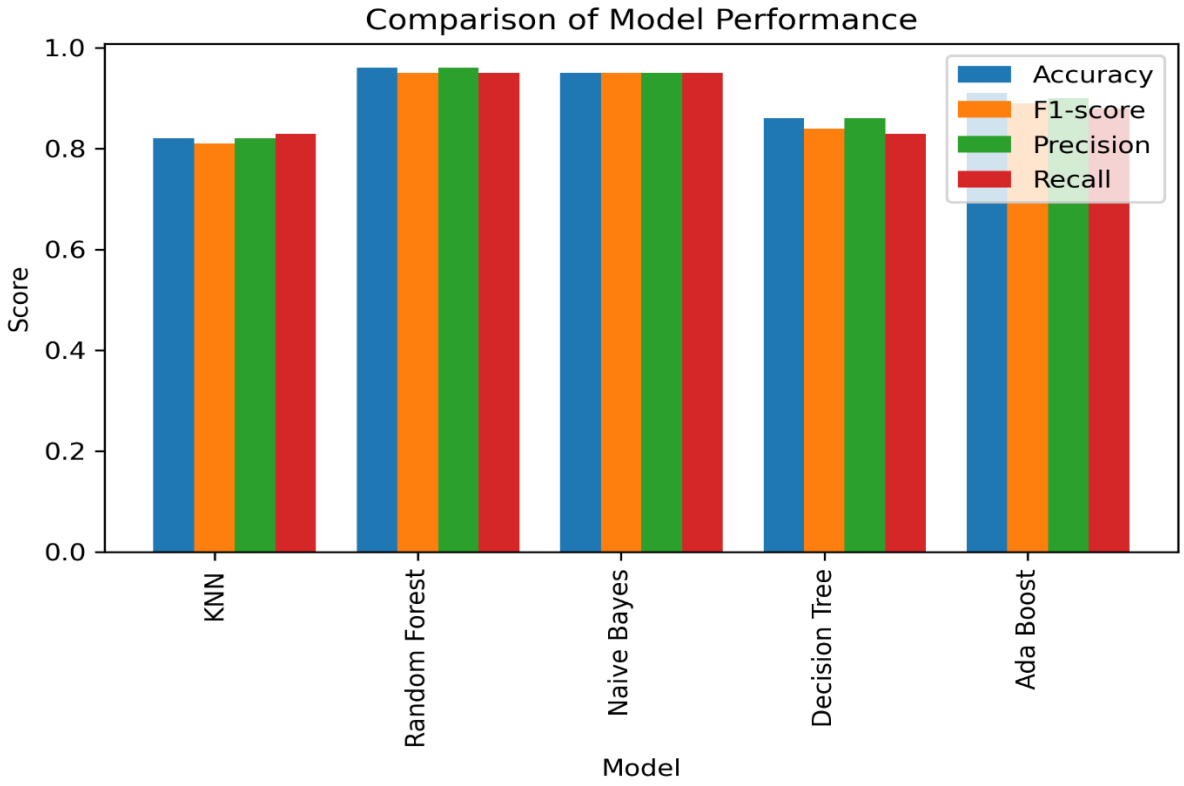

Figure12. Bar plot that displays comparison of each model performance, it is a graphical representation of the models' metrics, The statistical equation for this plot is not applicable, the (blue color) represents the accuracy, the (red color) represents the recall, the (green color) represents the precision, the (orange color) represents the F1-score, the value of each bar as shown in table6.

From the results, Naïve Bayes is the best model, these figures show the performance of the model:

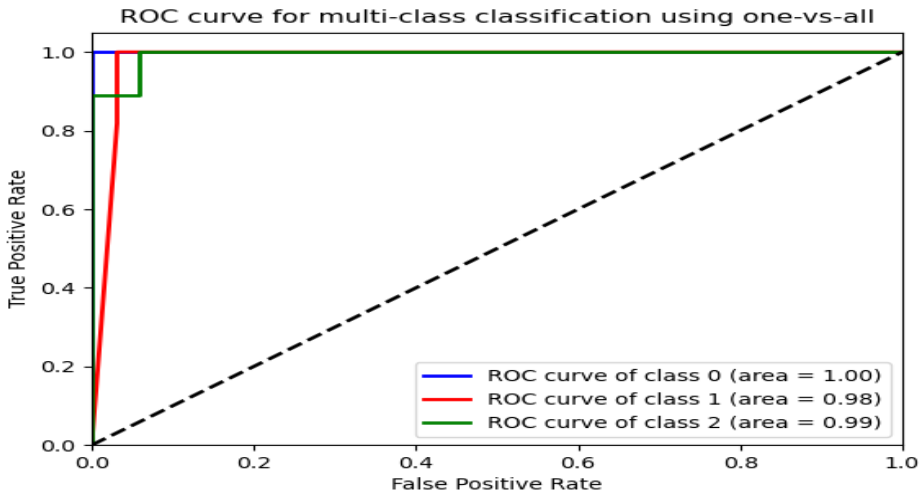

Figure13.Roc curve of Naïve Bayes.

The ROC (Receiver Operating Characteristic) curve is a graphical representation used to assess the performance of a classification model. It is commonly used in machine learning and statistics to evaluate the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity) at different classification thresholds. The ROC curve plots the true positive rate (TPR) on the y-axis against the false positive rate (FPR) on the x-axis. The TPR, also known as sensitivity or recall, represents the proportion of positive instances correctly classified as positive. The FPR represents the proportion of negative instances incorrectly classified as positive. To construct an ROC curve, the classification model's predictions are sorted according to their predicted probabilities or decision scores. The threshold for classification varies from low to high, and for each threshold, the TPR and FPR are calculated. The resulting points are then connected to form the ROC curve. The ideal ROC curve is a straight line from the origin to the top-left corner of the plot, representing a perfect classifier with no false positives and a high true positive rate. The area under the ROC curve (AUC-ROC) is often used as a summary measure of the model's performance. A higher AUC-ROC indicates better overall performance, with 1.0 representing a perfect classifier, and 0.5 representing a random classifier. The ROC curve and AUC-ROC provide valuable insights into the model's ability to discriminate between classes and help determine an appropriate classification threshold based on the desired balance between true positives and false positives.

The statistical equation for the ROC curve can be broken down into several components:

- True positive rate (TPR): The true positive rate is the proportion of positive cases that are correctly classified as positive by the classifier. It is also known as sensitivity.

- False positive rate (FPR): The false positive rate is the proportion of negative cases that are incorrectly classified as positive by the classifier. It is calculated as 1 - specificity, where specificity is the proportion of negative cases that are correctly classified as negative by the classifier.
- Discrimination threshold: The discrimination threshold is the threshold at which the classifier decides whether an observation is positive or negative. It is usually set to 0.5 but can be varied to evaluate the performance of the classifier at different levels of sensitivity and specificity.
- ROC curve: The ROC curve is a plot of TPR against FPR for different values of the discrimination threshold. Each point on the curve corresponds to a different value of the discrimination threshold.

As shown from the curve, the diagonal line is the line of no-discrimination or the line of chance. It represents the performance of naïve bayes, which has no ability to distinguish between positive and negative cases. As such, the diagonal line has an area under the curve (AUC) of 0.5, which is the lowest possible AUC value. The (blue line) represents the area of class 0 and it is equal to 1, the (red line) represents the area of class 1 and it is equal to 1, the (green line) represents the area of class 2 and it is equal to 0.99.

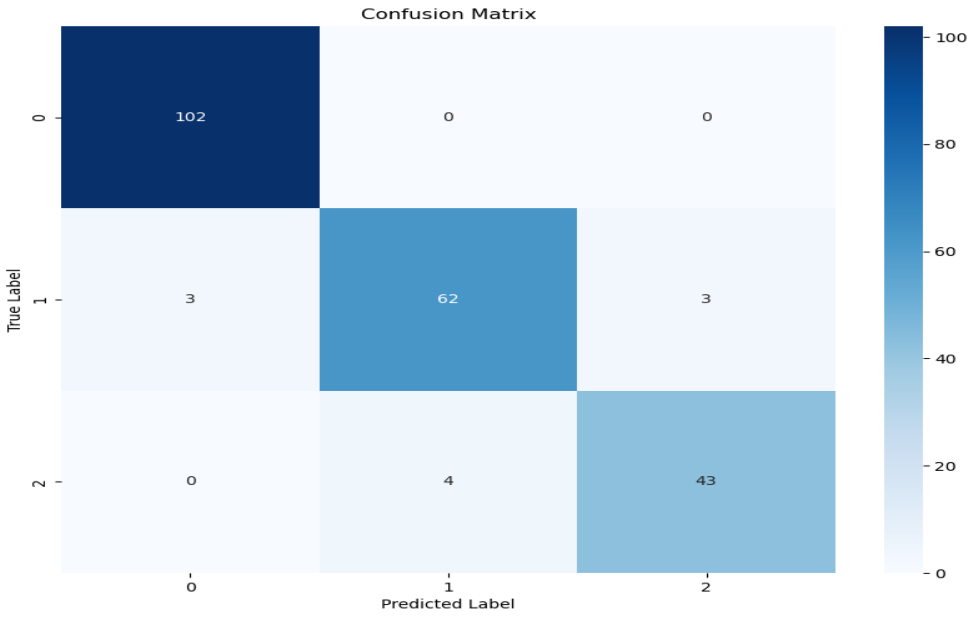

Figure14. Confusion matrix of Naïve Bayes, it shows the correctness of the predictions for the three classes, the malignant class was perfectly classified 102 of 102, the benign class about 62 sample of 68 was classified correctly and the healthy class about 43 sample of 47 was classified correctly.

The statistical equation for a confusion matrix can be broken down into several components:

- Actual class labels: The actual class labels of the observations are represented in the rows of the confusion matrix.
- Predicted class labels: The predicted class labels of the observations are represented in the columns of the confusion matrix.
- True positives (TP): True positives are the number of observations that are correctly classified as positive by the model. They correspond to the cases where the actual class label is positive, and the predicted class label is also positive.
- False positives (FP): False positives are the number of observations that are incorrectly classified as positive by the model. They correspond to the cases where the actual class label is negative, and the predicted class label is positive.
- True negatives (TN): True negatives are the number of observations that are correctly classified as negative by the model. They correspond to the cases where the actual class label is negative, and the predicted class label is also negative.
- False negatives (FN): False negatives are the number of observations that are incorrectly classified as negative by the model. They correspond to the cases where the actual class label is positive, and the predicted class label is negative.

Feature selection with Naïve Bayes and Random Forest

Trained random forest and naïve bayes on selected features before and after applying PCA,

10 features of 21 selected by Pearson correlation, and 7 features of 21 selected by Fisher correlation.

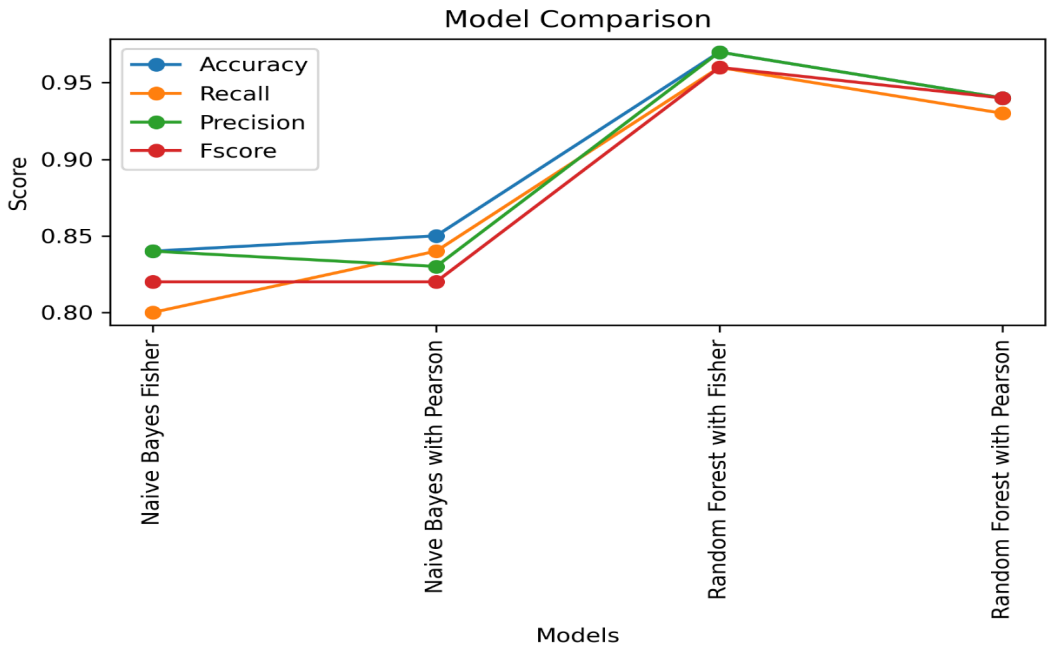

Figure15. Comparison of the best 2 models' performance after feature selection PCA analysis result using line plot, the (blue color) represents the accuracy, the (red color) represents the F1-score, the (green color) represents the precision, the orange color represents the recall.

The statistical equation for a line plot can be broken down into several components:

- Data points: The line plot represents a set of data points that are typically arranged in a two-dimensional space, with one variable on the x-axis and another variable on the y-axis.
- Line segments: The data points are connected by straight line segments, which represent the relationship between the variables.
- Axes: The line plot includes two axes, one for the x-variable and one for the y-variable. The axes are typically labeled with variable names and scaled to show the range of values in the data set.
- Tick marks: The axes are marked with tick marks to indicate the position of the data points along each axis.
- Legend: The line plot may include a legend that identifies the variables or data series represented by each line segment.

PCA analysis

Applied PCA after feature selection, for every selected feature, applied it with number of components = 5, The results showed that the set of the 5 selected features could keep the same level of prediction accuracy although 16 features were dropped.

Table.8 Result of random forest and naïve bayes on selected features using Pearson and fisher before PCA.

	Accuracy	F-score	Recall	Precision
Random forest + Pearson	0.94	0.94	0.93	0.94
Random forest + Fisher	0.97	0.96	0.96	0.97
Naïve Bayes + Pearson	0.85	0.82	0.83	0.84
Naïve Bayes + Fisher	0.84	0.82	0.8	0.84
Random forest + Pearson + PCA	0.87	0.84	0.84	0.84
Random forest + Fisher + PCA	0.89	0.87	0.86	0.89
Naïve Bayes + Pearson + PCA	0.8	0.78	0.76	0.81
Naïve Bayes + Fisher + PCA	0.82	0.8	0.78	0.84

From these results, Random Forest was better than naïve Bayes.

Random Forest + Pearson vs. Random Forest + Pearson + PCA: The addition of PCA reduced the accuracy, F-score, recall, and precision of the Random Forest model with Pearson features. This suggests that PCA may have removed some important information from the data, or that the original features were already well-suited for the model.

Random Forest + Fisher vs. Random Forest + Fisher + PCA: The addition of PCA slightly improved the accuracy and F-score of the Random Forest model with Fisher features, while slightly reducing the recall and precision. This suggests that PCA may have helped to reduce the complexity of the model and mitigate overfitting, without losing too much information.

Naïve Bayes + Pearson vs. Naïve Bayes + Pearson + PCA: The addition of PCA reduced the accuracy, F-score, recall, and precision of the Naïve Bayes model with Pearson features. This suggests that PCA may not be well-suited for this type of model, or that the original features were already optimal.

Naïve Bayes + Fisher vs. Naïve Bayes + Fisher + PCA: The addition of PCA slightly reduced the accuracy, F-score, and recall of the Naïve Bayes model with Fisher features, while slightly

improving the precision. This suggests that PCA may have helped to simplify the model and reduce noise, but at the cost of losing some important information.

Overall, the effect of PCA on the performance of the models is not consistent, and depends on the specific dataset, features, and model. In some cases, PCA can improve the performance by reducing noise and overfitting, while in other cases it can reduce the performance by removing important information or introducing bias. It is important to carefully evaluate the trade-offs between accuracy, interpretability, and complexity when using PCA in machine learning.

The ratio of variance explained in Principal Component Analysis (PCA) refers to the proportion of the total variance in the data that is accounted for by each principal component. PCA is a dimensionality reduction technique that transforms a set of correlated variables into a new set of uncorrelated variables called principal components. In PCA, the principal components are ordered in such a way that the first component explains the most variance in the data, followed by the second component, and so on. The ratio of variance explained is calculated by dividing the variance of each principal component by the total variance of the original data.

To compute the ratio of variance explained for a principal component, you take the eigenvalue associated with that component (which represents the variance) and divide it by the sum of all the eigenvalues (which represents the total variance). Mathematically, the ratio of variance explained for the i-th principal component can be expressed as:

(Variance explained by i-th component) / (total variance) = eigenvalue of i-th component / sum of all eigenvalues. The ratio of variance explained is typically presented as a percentage. It provides insight into the relative importance of each principal component in capturing the variability present in the data. Higher values indicate that the corresponding principal component explains a larger portion of the total variance, while lower values indicate a smaller contribution. By examining the ratio of variance explained for each principal component, you can determine the number of components needed to capture a desired amount of variance. Selecting a subset of principal components that explains a significant portion of the variance can help reduce the dimensionality of the data while preserving most of the important information.

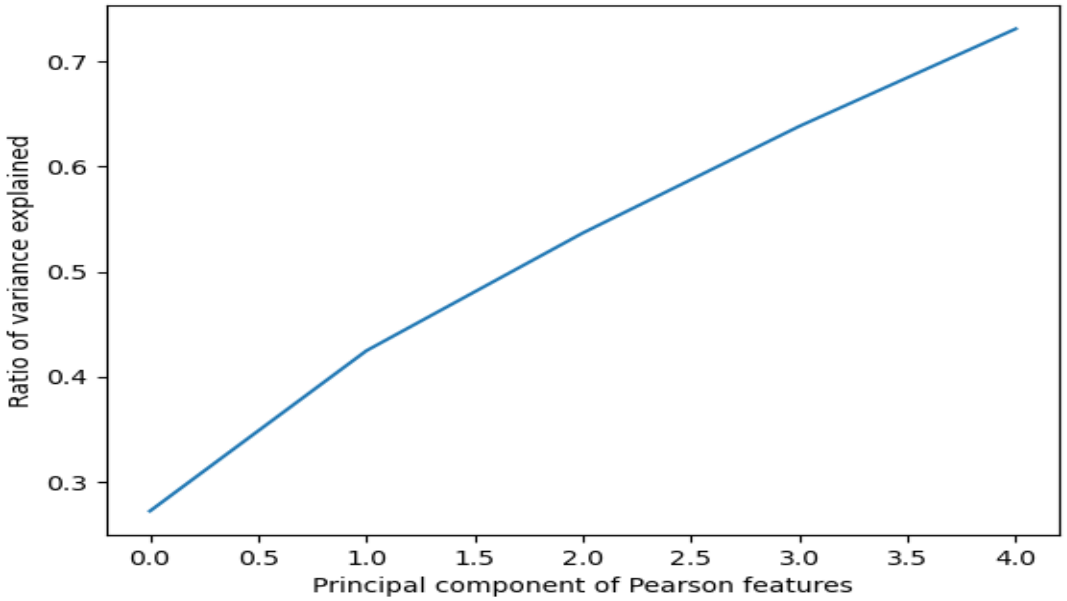

Figure16. The ratio of variance explained of PCA for features selected by Pearson, the ratio of variance explained by each principal component is calculated by dividing the variance of that principal component by the total variance of the original data set. This ratio represents the proportion of the total variance that is accounted for by that principal component and it equal to 0.4. The sum of the ratios of variance explained for all the principal components is always equal to 1.

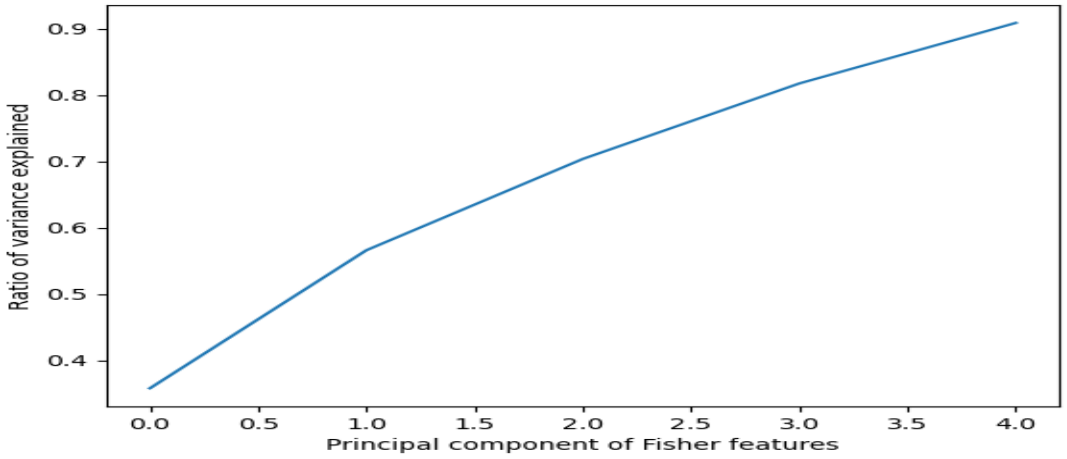

Figure17. Ratio of variance explained of PCA for features selected by Fisher. It's statistical equation as shown in Figure.16, This ratio represents the proportion of the total variance that is accounted for by that principal component and it equal to ~0.6.

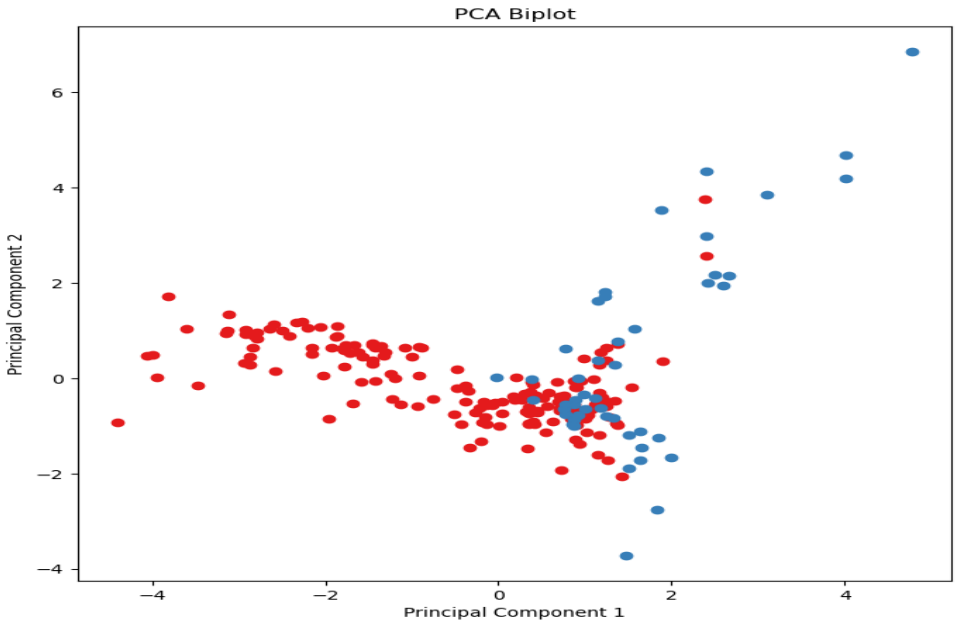

Figure18. Biplot shows the transformed data in the space of the first two principal components for features selected by Pearson. The (red color) represents the first principal component, and the (blue color) represents the second principal component.

The statistical equation for a biplot can be broken down into several components:

- Data matrix: A biplot is based on a data matrix that contains measurements for multiple variables on multiple observations. The data matrix is typically standardized to have a mean of zero and a standard deviation of one for each variable.
- Singular value decomposition (SVD): The data matrix is decomposed using SVD, which factorizes the matrix into three components: a matrix of left singular vectors, a diagonal matrix of singular values, and a matrix of right singular vectors. The left singular vectors represent the observations, and the right singular vectors represent the variables.
- Scaling: The singular values are scaled by the square root of the number of observations in the data set, which ensures that the sum of squares of the singular values equals the total variance of the data.
- Plotting: The left and right singular vectors are plotted in the same two-dimensional space, with the observations and variables represented as points. The distance between the points represents the correlation between the observations and variables.
- Annotation: The biplot is annotated with labels for the observations and variables to aid interpretation.

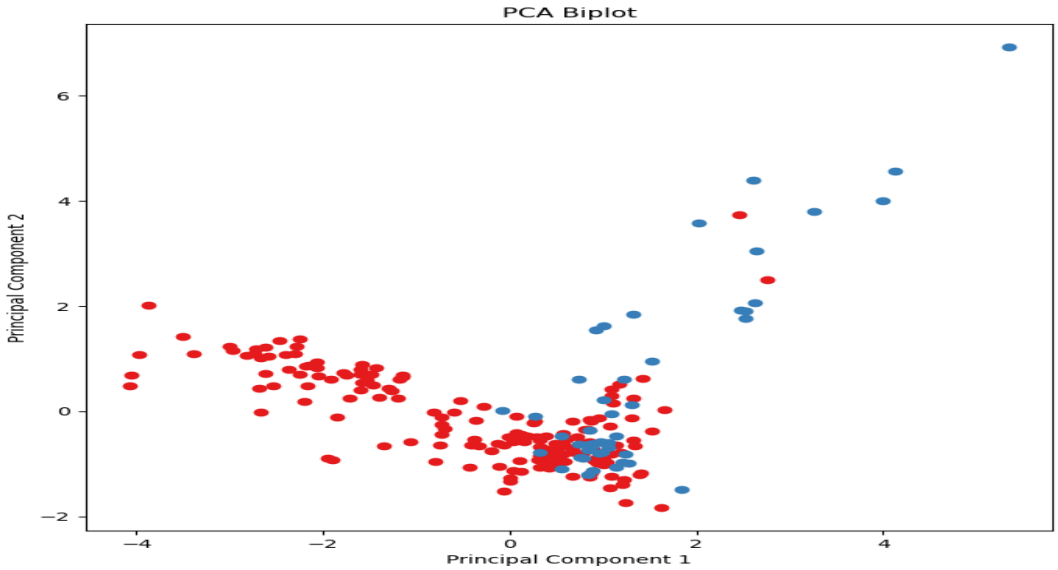

Figure19. Biplot shows the transformed data in the space of the first two principal components for features selected by Fisher. The statistical equation as shown in Figure18. The (red color) represents the first principal component, and the (blue color) represents the second principal component.